Please rewrite claim 9 as shown.

9. (Amended) A register definition file stored on a computer readable medium and comprising a plurality of register definition tables, each table including at least predefined sectors for the bit location within a register of an element, the name of the element, the function of the element and the functional status of the element, and each table further including the word location of the register within a memory map, wherein when the computer readable medium is loaded into a computer for simulating an Application Specific Processor, said register definition tables are accessed based on said word location for simulation of said registers.

REMARKS

In response to the Office Action mailed July 5, 2002, Applicant respectfully requests reconsideration. To further the prosecution of this application, amendments have been made to the specification and the claims, and the claims as presented are believed to be in allowable condition.

Attached hereto are marked-up versions of the changes made to the specification and to the claims by the current amendment. The attached pages are captioned **"MARKED-UP SPECIFICATION"** and **MARKED-UP CLAIMS,"** respectively.

Applicant notes the objection in the Office Action to Figure 7, asserting that the phrase "STATE PERIPUCAL" is incorrect. Applicant will file a corrected Figure 7, wherein the phrase "STATE PERIPUCAL" will be changed to "STATE PERIPHERALS", upon allowance of the application.

The Office Action objected to the specification, asserting that informalities exist. Applicant has amended the specification to correct the noted informalities.

The Office Action objected to the specification for containing a computer program listing consisting of more than ten pages. The Office Action asserted that the computer program listing must be submitted as a microfiche appendix and must be appropriately referenced in the specification. Applicant notes that 37 CFR §1.96 permits the submission of computer program listings on a compact disc. Accordingly, Applicant has submitted a compact disc containing the computer program listing appearing pages 17-31 of the specification. Applicant has also amended the specification to remove the text of the computer program listing and to appropriately reference the compact disc appendix. Accordingly, it is respectfully requested that the objection to the specification in paragraph 5 of the Office Action be withdrawn.

The Office Action objected to claim 5 under 37 CFR §1.75(d)(1), asserting that the phrase "allocating specific elements...to predefine sectors is incorrect." Applicant has amended claim 5 to overcome this objection.

The Office Action rejected claim 9 under 35 U.S.C. §101, asserting the claim recites nonfunctional descriptive material. Applicant respectfully disagrees with this rejection. As stated in MPEP §2106 IV(B)(1)(a), "a claimed computer-readable medium encoded with a data structure defines structural and functional interrelationships between the data structure and the computer software and hardware components which permit the data structure's functionality to be realized, and is thus statutory." The register definition file of claim 9, as amended, recites functional and structural interrelationships between itself and the computer software and hardware components that permit its functionality to be realized. For example, claim 9 recites that "when the computer readable medium is loaded into a computer for simulating an Application Specific Processor (ASP), said register definition tables are accessed based on said word location for simulation of said registers." Accordingly, it is respectfully requested that the rejection of claim 9 under 35 U.S.C. §101 be withdrawn.

The Office Action rejected claims 1-9 under 35 U.S.C §103(a) as obvious over Aleksic (5,995,736) in view of Shimabukuro (5,557,774). Applicant respectfully traverses this rejection.

Aleksic is directed to a method and system for automatically modelling registers for integrated circuit design. The system for automatically modelling registers includes a model register generator 36 which accesses common register description source data (col. 5, lines 19-22). Model register generator 36 reads in the source data and generates a number of outputs (col. 5, lines 37-39). One output generated by the model register generator is a coded register layer that defines characteristics of register block decoders and associated registers. The coded register layer includes register code templates which are in a low-level hardware description language such as VHDL (col. 6, lines 15-25).

Shimabukuro is directed to developing test programs for testing software of a plurality of different subsystems of a computer system. Shimabukuro discloses preparing an application-specific hardware simulator to simulate the computer system and then running test programs on the simulator (col. 6, lines 1-12). Thus, Shimabukuro is directed to creating a software simulation of hardware that has already been implemented for the purposes of testing software. Aleksic, on the other hand, is directed to creating a software simulation of hardware that is not yet implemented, for the purposes of verifying that the hardware design is accurate.

With regard to claim 1, the rejection is improper because the Office Action failed to establish a *prima facie* case of obviousness. MPEP §2142 states three criteria that must be met to establish a *prima facie* case of obviousness. First, there must be some suggestion or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art, to modify the reference or to combine reference teachings. Second, there must be a reasonable expectation of success. Finally, the combined prior art references must teach or suggest all the claim limitations. The teaching or suggestion to make the claimed combination and the reasonable expectation of success must both be found in the prior art, and not based on applicant's disclosure (MPEP §2142, page 2100-121, Eighth Edition). Applicant respectfully submits that the Office Action does not meet any of these three requirements.

First, the Office Action does not point to any disclosure or suggestion in the references that would motivate one of ordinary skill in the art to modify or combine the references. The Office Action merely states that it would have been obvious to modify the computer program of Aleksic with the computer program of Shimabukuro because such a modification would facilitate "modeling and simulating the behavior of an actual computer system with peripherals." However, as discussed above the Aleksic and Shimabukuro references are directed to two completely different problems.

Aleksic is directed to developing a software model to simulate an ASIC that is not yet implemented in hardware. Such a simulation may help a designer to ensure that the hardware ASIC will operate properly, without having to go through the costly process of implementing the ASIC in silicon. By contrast, Shimabukuro is directed to simulating a specific computer system in software to help a software developer test software for the computer system without having to actually configure a hardware computer system that meets the desired specifications. Because Aleksic is directed to testing an ASIC and Shimabukuro is directed to testing software, one of ordinary skill in the art would not be motivated to combine the two references as they are directed to solving different problems. Further, there is no disclosure or suggestion in Shimabukuro that using an input file relating to a description of a peripheral would have been useful in designing an ASIC. Such a teaching or suggestion comes from Applicant's own disclosure and not from the prior art.

Second, there must be a reasonable expectation of success. The Office Action does not discuss how the computer program of Aleksic and the computer program of Shimabukuro would have been combined successfully, let alone combined successfully in a manner that would obviate Applicant's claimed invention. Nonetheless, it appears that such a combination would

not have been successful. First, in combining the two computer programs, the Office Action infers that the inputs described in Shimabukuro would replace the register description source data of Aleksic. However, the inputs described by Shimabukuro are specified in a hardware description language such as HDL (Shimabukuro, Col. 6, lines 46-59). In contrast, the Model Register Generator of Aleksic is designed to process text input, not HDL. Thus, the combination would not work as the Model Register Generator of Aleksic would not be able to process the HDL input of Shimabukuro. Further, Aleksic discloses using the Model Register Generator to automatically generate an HDL description of registers based on text input (i.e., the source data). If the HDL input of Shimabukuro were used, then the Model Register Generator of Aleksic is useless because it is unnecessary to generate an HDL description if the description is already specified by the input of Shimabukuro. Therefore, there is no reasonable expectation of success in combining the two references.

Lastly, even if one were to combine the two references, the references, alone or in combination, would not teach all of the limitations of Applicant's claim 1. Claim 1 recites, "generating for each peripheral an input file which defines the functional attributes of that peripheral in a high level language with an input data structure." The Office Action admits that no such input file is disclosed or suggested by Aleksic. Further, Shimabukuro also fails to disclose or suggest such an input file. The inputs disclosed by Shimabukuro are provided in a low-level hardware description language such as HDL. By contrast, claim 1 requires that the input file be in a high level language.

Additionally, claim 1 recites "entering the input file into the computer system and operating a modelling tool loaded on the computer system to generate from the input file a register definition file by allocating specific elements of the input data structure to predefined sectors of a register definition table." Neither Aleksic nor Shimabukuro disclose allocating elements from the input data structure to a predefined register definition table. Such a table is not disclosed or suggested by either reference, taken alone or in combination. Accordingly, it is respectfully requested that rejection of claim 1 under 35 U.S.C. §103(a) be withdrawn.

Claims 2-7 depend from claim 1 and are patentable for at least the same reasons discussed above in connection with claim 1. Accordingly, it is respectfully requested that the rejection of claims 2-7 under 35 U.S.C. §103(a) be withdrawn.

With regard to claims 6 and 7, for which Official Notice was taken, Applicant does not dispute that a computer system, such as that disclosed in Figure 4 of Aleksic, might include an input means and an output means, despite the fact that neither is literally shown. However, claim

6 recites that the input means comprises means for "receiving a physical recording device holding the input file for each peripheral." As no such input file is disclosed or suggested by either of the two references, Applicant respectfully traverses the rejection of claim 6. Similarly, claim 7 recites that the output means comprises means for "loading the register definition file onto a physical recording device." Because such a register definition file is not disclosed or suggested in the references, Applicant respectfully traverses the rejection of claim 7. Pursuant to M.P.E.P. §2144.03, the Examiner is requested to cite a reference in support of these rejections, or withdraw the rejections of claim 6 and 7.

Claims 8 and 9 are also rejected under 35 U.S.C. §103(a) over Aleksic in view of Shimabukuro. However, as discussed above in connection with claim 1, the Aleksic and Shimabukuro references are not combinable. Accordingly, it is respectfully requested that the rejection of claims 8 and 9 under 35 U.S.C. §103(a) be withdrawn.

CONCLUSION

In view of the foregoing amendments and remarks, this application should now be in condition for allowance. A notice to this effect is respectfully requested. If the Examiner believes after this Amendment that the application is not in condition for allowance, the Examiner is requested to call Applicant's attorney at the number listed below to discuss any outstanding issues relating to allowability.

If this response is not considered timely filed, and if a request for an extension of time is otherwise absent, Applicant hereby requests any necessary extension of time. If there is a fee occasioned by this response, including an extension fee, that is not covered by the enclosed check, please charge any deficiency to Deposit Account No. 23/2825.

Respectfully submitted,

Gajinder Singh PANESAR

Robert A. Skrivanek, Jr.Reg. No.: 41,316
Wolf, Greenfield & Sacks, P.C.
600 Atlantic Avenue
Boston, MA 02210-2211
(617)720-3500

Date: December 5, 2002

MARKED UP SPECIFICATION

Please replace the paragraph beginning at line 26 of page 2 as shown.

the processor being operable to execute the program representing the modelling tool to generate from the input file a register definition file by allocating specific elements of the input data structure to [predefine] predefined sectors of a register definition table; and

Please replace the paragraph beginning at line 1 of page 7 as shown.

An application specific processor is modelled as a central processor (CPU) 2 and a set of peripherals 4. The CPU 2 is modelled with the basic elements of an interrupt handler 6 and memory 8. A set of applications running on the CPU are denoted by the process circle 10 labelled APPLS. Each peripheral 4 is modelled with an internal interface 12 between the peripherals and the CPU 2 and an external interface 14 between the peripheral and the "outside world", that is [externally] external of the ASP. At the time of modelling the ASP, it is not known whether or not the peripherals will in fact be implemented in software, hardware or some combination of both. However, whether finally implemented in software or hardware or some combination of both, the peripherals 4 represent how the central processor 2 cooperates with the external environment. The external interfaces 14 receive stimuli S from the external environment and generate responses R in response to the stimuli. These are carried by the external interface 14. The internal interfaces 12 carry state information and data between the peripherals and the applications 10 running on the CPU 2. This is described in more detail with reference to Figure 2.

Please replace the paragraph beginning at line 16 of page 8 as shown.

An input file is created for each peripheral 4 in a high level language such as C using an input data structure compatible with that language. That input file defines the interface behaviour of the peripheral 4 with respect to the CPU. The architect determines the responses R of the peripheral with respect to external stimuli S. A modelling tool 24 generates automatically from the data structure defined in the input file [24] 22 a documentation file 26, an interface functions file 28, and a test functions file 30.

Please replace the paragraph beginning at line 24 of page 9 as shown.

The test functions [take the form] define the attributes of the CPU read/write paths 18, 20 and include:

Please replace the paragraph beginning at line 13 of page 11 as shown.

word offset – defining an offset location of the register in memory

bit offset - defining the bit location of each element in the register and derivable from the bit length BL in the data structure

bit field – naming the element of that bit location

function – defining the function F of the element

reset state – value of entity on reset

read/write – whether entity is [read] readable or writable from CPU

Please replace the paragraph beginning at line 12 of page 12 as shown.

Figure 5 is a flow chart illustrating high level operation of the modelling tool 24. At step S1, input parameters given to the modelling tool 24. At step S1, input parameters give to the modelling tool are checked. At step S2, the input file corresponding to one of the specified parameters is opened. Its contents are checked and any [areas] errors are reported in a meaningful manner (step S3) by an error routine. If the contents of the input file are valid, the files are opened and named at step S4. At step S5, the files are created as described earlier with reference to Figures 3 and 4. Finally, at step S6, the files are closed.

MARKED UP CLAIMS

5. (Amended) A computer system which comprises a processor and a memory, the memory holding a program representing a modelling tool for use in designing an application specific processor (ASP), wherein the computer system comprises an input means for receiving a plurality of input files, each input file defining the functional attributes of a peripheral for the ASP in a high level language within an input data structure;

the processor being operable to execute the program representing the modelling tool to generate from the input file a register definition file by allocating specific elements of the input data structure to [predefine] predefined sectors of a register definition table; and

wherein the computer system further comprises an output means for outputting the register definition file in a manner which is usable to create in silicon the registers of the ASP.

9. (Amended) A register definition file stored on a computer readable medium and comprising a plurality of register definition tables, each table including at least predefined sectors for the bit location within a register of an element, the name of the element, the function of the element and the functional status of the element, and each table further including the word location of the register within a memory map, [for access during simulation of an ASP implementing the registers] wherein when the computer readable medium is loaded into a computer for simulating an Application Specific Processor, said register definition tables are accessed based on said word location for simulation of said registers.